RAMS
Reliability, Availability,
Maintainability, and Safety

Compering the ACER and POT MCMC (acronyms?) Extreme Value Statistics Methods Through Analysis of Commodities Data

Kristoffer Kofoed Rødvei

April 2016

MASTER THESIS

Department of Production and Quality Engineering (What here??)

Norwegian University of Science and Technology

Supervisor: Professor Arivd Næss

Secondary Advisor: Professor Andre Riebler

Helping(???): Professor Sjur Westgaard

Contents

Chapter 1

Theory

The following chapter gives an introduction to the theory behind this work. Some of the sections only give a brief description of the theory with sources for more in depth explanation.

1.1 Extreme value theory

X random variable x observed! The basics of extreme value theory, is to analyze the maximum of a series of random variables $X_1, \ldots, X_n$. Defining M_n as the maximum of a series

$$M_n = \max\{X_1, \ldots, X_n\}, \tag{1.1}$$

the resulting distribution of M_n is

$$\Pr(M_n \leq z) = \Pr(X_1 \leq z, \ldots, X_n \leq z). \tag{1.2}$$

By assuming $X_1, \ldots, X_n$ independent and identically distributed (i.i.d.) with common cumulative distribution function F, equation (1.2) reduces to

$$\begin{aligned} \Pr(M_n \leq z) &= \Pr(X_1 \leq z) \times \cdots \times \Pr(X_n \leq z) \\ &= \left[F(z)\right]^n. \end{aligned} \tag{1.3}$$

The distributing F is normally unknown, and a divergent in the estimated distributing F could potentially escalate to a large difference in the resulting F^n.

As for the central limit theory for the normal distribution, F^n also has a limiting distribution, by The Fisher–Tippett–Gnedenko theorem. If there exist an $a_n > 0$ and b_n such that

$$\lim_{n\to\infty} \Pr\left[(M_n - b_n)/a_n \leq z\right] \to G(z), \tag{1.4}$$

where G is a non-degenerating function, then G follows either

$$G(z) = \exp\left\{-\exp\left[-\left(\frac{z-b}{a}\right)\right]\right\}, \qquad -\infty < z < \infty; \tag{1.5}$$

$$G(z) = \begin{cases} 0, & z \leq b, \\ \exp\left\{-\left(\frac{z-b}{a}\right)^{-\alpha}\right\}, & z > b; \end{cases} \tag{1.6}$$

$$G(z) = \begin{cases} \exp\left\{-\left[-\left(\frac{z-b}{a}\right)\right]^{\alpha}\right\}, & z \leq b, \\ 1, & z > b; \end{cases} \tag{1.7}$$

for each case, $\alpha > 0$. Here equation (1.5), (1.6) and (1.7) refears to Gumbel, Fréchet and Weibull distribution respectively. The above equation can be combined into the General Extreme Value (GEV) distribution

$$G(z) = \exp\left\{-\left[1+\xi\left(\frac{z-\mu}{\sigma}\right)\right]^{-\frac{1}{\xi}}\right\} \tag{1.8}$$

where $1+\xi(z-\mu)/\sigma > 0$, the location parameter is $-\infty < \mu < \infty$, the scale parameter is $\sigma > 0$ and the shape parameter is $-\infty < \xi < \infty$. It can easily be verified that the GEV equals equation (1.6) when $\xi > 0$, equation(1.7) when $\xi < 0$ and converge towards equation(1.5) as $z \to 0$.

The GEV model requires i.i.d. data points for parameter estimations. Unfortunately, in practice that is rarely the case. Block maxima or r largest order statistics are common methods for filtering the dependent data points into an approximate i.i.d. dataset (Coles, 2001, p. 66). The basic principle is to only use the largest, or r largest data within each block. Examples of block sizes could be week, month, year etc.

For a deeper description of extreme value theory, GEV or block maxima, the book of (Coles, 2001, Chapter 3) is suggested.

1.1.1 Peak Over Threshold

One of the problem with the GEV method and the i.i.d. filtration of data points, is that the block maxima and r largest order statistics are quite wasteful. Specially in situations where some blocks contains more extreme than others, large extreme will be discarded from the set, which else would have been accepted in other blocks.

The Peak Over Threshold (POT) method is suggesting a different method of tackling the i.i.d. filtration. Filtering the data, by only using points over a certain threshold u, avoid the problem of discarding large extremes in certain blocks. As long as u is chosen sufficiently large, the resulting data will be i.i.d. Using the GEV distribution equation (1.8) it can be shown, like was done by (Coles, 2001, p. 76), that

$$\Pr(X > y+u | X > u) = \left(1 + \frac{\xi y}{\tilde{\sigma}}\right)^{-\frac{1}{\xi}}, \tag{1.9}$$

where y is the threshold excess, given by $y = z - u$ for $z > u$. The resulting distribution of y is called the generalized Pareto distribution (GPD)

$$H(y) = 1 - \left(1 + \frac{\xi y}{\tilde{\sigma}}\right)^{-\frac{1}{\xi}} \tag{1.10}$$

or

$$H(y) = 1 - \exp\left(-\frac{y}{\tilde{\sigma}}\right) \tag{1.11}$$

when $\xi \to 0$. Here ξ equals the GEV parameter, while $\tilde{\sigma} = \sigma + \xi(u - \mu)$. The conditional probability

$$\begin{aligned}\Pr(X > y+u | X > u) &= \frac{\Pr(X > y+u, X > u)}{\Pr(X > u)} \\ &= \frac{\Pr(X > y+u)}{\Pr(X > u)},\end{aligned} \tag{1.12}$$

since $y + u \geq u$, the probability $\Pr(X > y+u, X > u) = \Pr(X > y+u)$. By combining equation

(1.9) and (1.12), the probability of a future event can be found by

$$\begin{aligned}\Pr(X > z) &= \Pr(X > y + u)\\ &= \Pr(X > u)\cdot P(X > y + u|X > u)\\ &= \Pr(X > u)\cdot\left[1+\xi\left(\frac{z-u}{\tilde{\sigma}}\right)\right]^{-\frac{1}{\xi}},\end{aligned} \tag{1.13}$$

where $\Pr(X > u)$ is the probability that a random point exceeds the threshold.

For more information about the POT method, see (Coles, 2001, Chapter 4).

ADD/TALK/NAME GENERALIZED PARETO DISTRIBUTION (GPD) (H(y)).

y=x-u CALLED THRESHOLD EXCESS. INCLUDE IN OTHER PARTS OF THE THESIS!

Declustering

The numbers of threshold excess y increase as the threshold u decrease. A larger number of threshold excess will increase the accuracy and lower the variance of the parameter estimation, which suggest using a low threshold. In practice, data are often correlated, heteroscedastic or nonstationary. For data without trend, a high enough threshold will ensure close to i.i.d. property for the threshold excess. As the threshold decrease, clusters could appear, and the threshold excess will no longer be i.i.d. Violation of the i.i.d. property will result in an estimation bias, which suggest using a high threshold. The selection of threshold comes down to the trade-off between accuracy and bias. The goal is to get the lowest variance without bias.

Declustering method can be applied to improve the i.i.d. property for low threshold. The target is to localize clusters above the threshold and select the largest value within each cluster. The method used here, defines a cluster as the points above the threshold, until r consecutive points are observed below. Referring to figure 1.1, for an example on how the method is used in practice. For $r = 1$ there are 7 clusters, while for $r = 4$ there are 3. For the particular threshold used in the plot, $r = 4$ seems like the superior choice.

For a more in depth description of decluttering, see (Coles, 2001, p. 100).

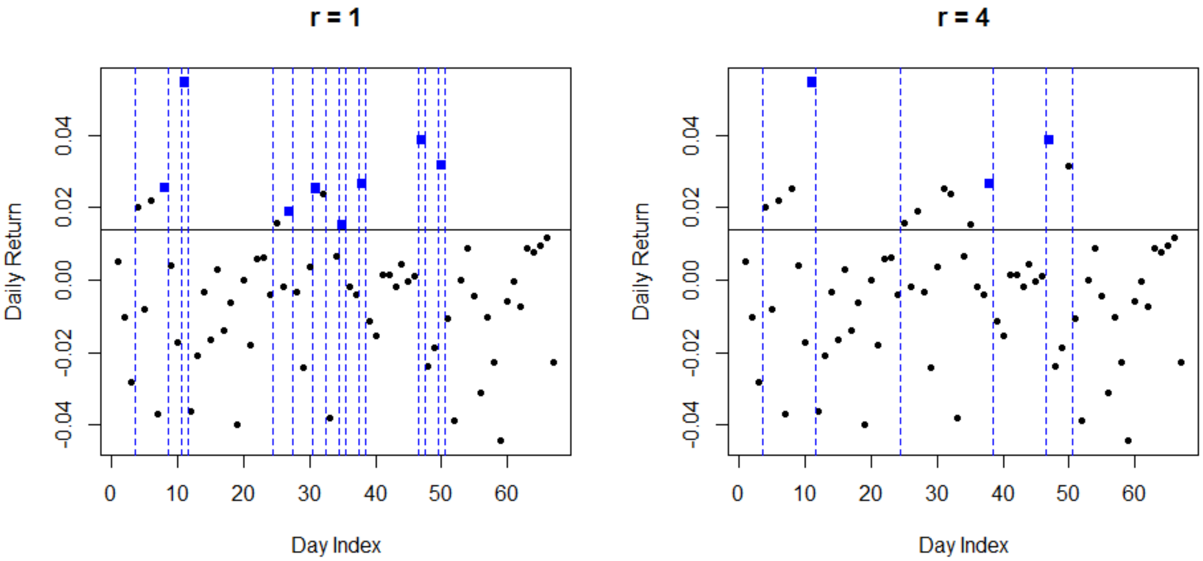

Figure 1.1: Portion of the crude oil daily return series, described in chapter 2.2. The horizontal solid line is threshold, with $u = 0.014$. Clusters are localized between the vertical blue dashed lines, with the largest value within each clusters shown as a blue square.

Threshold

As stated above, the goal when selecting threshold u is to find the smallest threshold u_0, for which the model is still unbiased, such that the highest accuracy is achieved. For this paper, the combination of two methods are used in the selection process.

The first method uses the fact that the mean of the GPD equals

$$E(Y) = \frac{\tilde{\sigma}}{1-\xi}, \tag{1.14}$$

for $\xi < 1$, and infinite when $\xi > 1$. Thus the first method fails when $\xi > 1$, but in practice ξ rarely exceeds 1. As shown above, the GPD ξ equals the GEV parameter which is independent of threshold, while $\tilde{\sigma} = \sigma + \xi(u - \mu)$ is linear with respect to threshold. Here σ and μ are the GEV parameters and independent of threshold. Thereby the mean of Y is also linear proportional to threshold. By plotting the mean of threshold excess against thresholds, linear effect should be apparent from u_0. Confidence interval can be added for a better understanding of where the linearity starts. For larger values of thresholds, there will only be a few numbers of threshold excess, hence it is suggested using t-distribution for more realistic confidence intervals. REFERING TO

PLOT BLABLABLA. For more information about the method, see (Coles, 2001, p. 79).

For the second method, estimates of ξ and $\tilde{\sigma}$ is taken for a variety of thresholds. The parameters ξ and the reparametrized $\sigma^* = \tilde{\sigma} - \xi u$ should both be constant from u_0. For pinpointing of u_0, both ξ and σ^* is plotted against u, with added confidence intervals. REFERING TO PLOT BLABLABLA. For more information about the method, see (Coles, 2001, p. 83).

After threshold selection, $\tilde{\sigma}$ is simply estimated from the threshold excess and is required larger than zero. For simplicity, from here, the notation σ is used for the GPD parameter $\tilde{\sigma}$, as long as otherwise is not specified.

1.1.2 Average Condition Exceedance Rate

The Average Conditional Exceedance Rate (ACER) is another extreme value method, first introduced by Næss and Gaidai (2009), see the paper and Næss et al. (2013) for a more in depth description of the theory, as the following is only a brief introduction. Both the GEV and GPD distributions requires the observations to be i.i.d. When observations are not i.i.d., filtering methods such as, threshold exceedance, declustering, blocking, etc., is used to achieve close to i.i.d. data. The problem with these filtering methods, is that they often discard most of the data, such that only a small amount of the data can be used for parameter estimation. The advantage of the ACER method is that the observation is not restricted to i.i.d. or even stationarity data, only requires no trend. Another advantage is the ACER methods ability to a certain extent capture the subasymtotic parts, which also could improve estimation.

Without the i.i.d. assumption for $X_1, \ldots, X_n$, equation (1.2) can be written using time dependency

$$\Pr(M_n \le z) = \prod_{j=1}^{n} \Pr(X_j \le z, |X_{j-1} \le z, \ldots, X_1 \le z) \cdot \Pr(X_1 \le z). \tag{1.15}$$

It is reasonable to assume that the data dependency with neighboring points degrease by time, and is negligible after $k \ll n$ steps, such that

$\Pr(X_j \le z, |X_{j-1} \le z, \ldots, X_1 \le z) \approx \Pr(X_j \le z, |X_{j-1} \le z, \ldots, X_{j-k+1} \le z)$ for every $j = k, \ldots, n$. Using this and tailor expansion of the exponential function around zero, equation (1.15) reduces to

$$\begin{aligned}\Pr(M_n \le z) &\approx \exp\left(-\sum_{j=k}^{n} \alpha_{kj}(z) - \sum_{i=1}^{k-1} \alpha_{ii}(z)\right)\\ &\approx \exp\left(-\sum_{j=k}^{n} \alpha_{kj}(z)\right) \qquad (1.16)\end{aligned}$$

where $\alpha_{kj}(z) = \Pr(X_j \ge z | X_{j-1} \le z, \dots, X_{j-k+1} \le z)$ for $k \ge 2$ and $\alpha_{kj}(z) = \Pr(X_j \ge z)$ for $k = 1$. The final step is justified since $\sum_{i=1}^{k-1} \alpha_{ii}(z)$ is negligible compared to $\sum_{j=k}^{n} \alpha_{kj}(z)$ for $k \ll n$, while the tailor expansion around zero is reasonable at the upper tail since for large z, $\alpha_{kj}(z)$ is close to zero.

Considering the Average Conditional Exceed Rate (ACER) as

$$\epsilon_k(z) = \frac{1}{n-k+1} \sum_{j=k}^{n} \alpha_{kj}(z). \qquad (1.17)$$

The ACER function can be estimated using

$$\hat{\epsilon}_k(z) = \frac{\sum_{j=k}^{n} \mathbf{1}(x_j \ge z, x_{j-1} \le z, \dots, x_{j-k+1} \le z)}{\sum_{j=k}^{n} \mathbf{1}(x_{j-1} \le z, \dots, x_{j-k+1} \le z)}. \qquad (1.18)$$

where $\mathbf{1}(\omega)$ is the indicator function for event ω. For nonstationary observations it is suggested using $n-k+1$ as an approximation for the denominator. The approximation can be justified since $\mathbf{1}(x_{j-1} \le z, \dots, x_{j-k+1} \le z) \to 1$ in the upper tail where z is large.

It is assumed that the tail of the ACER function follows

$$\epsilon_k(z | a_k, b_k, c_k, q_k, \xi_k) = q_k \left[1 + \xi_k \left(a_k (z - b_k)^{c_k}\right)\right]^{-1/\xi_k}, \qquad (1.19)$$

where the parameters a_k, b_k, c_k, q_k and ξ_k are approximately constant in the upper tail for a certain k. The process of selecting k can be done by investigating the plot of $\hat{\epsilon}_k(z)$ against z for a verity of k, k is set to the smallest value for which increasing k makes negligible change to the tail. see figure REF TIL FIGUR PLUS SOME TEXT HER!!!!!!!!!!. The parameters can be estimated by minimizing the weighted square error

$$F(a,b,c,q,\xi)=\sum_{i=1}^{N} w_i\left[\log(\hat{\epsilon}_k(z_i))-log(q)+\xi^{-1}\log\left(1+a(z_i-b)^c\right)\right]^2, \tag{1.20}$$

using numerical methods. Selecting $z_1,\ldots,z_N$ is done by uniformly dividing the values from where regular tail behavior of $\hat{\epsilon}_k(z)$ starts to $\max_{1\le i\le n}(X_i)$ into N points. The weights w_i is calculated using

$$w_i=\left(\log\left[C_\alpha^+(z_i)\right]-\log\left[C_\alpha^-(z_i)\right]\right)^{-2}, \tag{1.21}$$

where $C_\alpha^+(z_i)$ and $C_\alpha^-(z_i)$ is the upper and lower $100\cdot\alpha\%$ confidence interval values respectively for $\hat{\epsilon}_k(z_i)$. Organizing the observation into R similar realizations, like R years, the sample variance can be calculated as

$$\hat{s}_k(z_i)^2=\frac{1}{R-1}\sum_{r=1}^{R}\left(\hat{\epsilon}_k^{(r)}(z_i)-\hat{\epsilon}_k(z_i)\right) \tag{1.22}$$

where $\hat{\epsilon}_k^{(r)}(z_i)$ is the estimated ACER function for the r realization at z_i. Hence a $100\cdot\alpha\%$ confidence interval can be calculated using the student t-distribution

$$C_\alpha^\pm(z_i)=\hat{\epsilon}_k(z_i)\pm t_{(1-\alpha)/2,R-1}\frac{\hat{s}_k(z_i)}{\sqrt{R}} \tag{1.23}$$

where $t_{p,v}$ is defined as $\Pr(T>t_{p,v})=p$ for the standardized t distribution with v degrees of freedom.

After parameter estimation, future prediction can be achieved using equation (1.19). Confidence intervals can be added to the ACER function prediction by estimating the parameters to the upper and lower confidence curve. Using $\epsilon_k(z_i|a,b,c,q,\xi)\pm t_{(1-\alpha)/2,R-1}\frac{\hat{s}_k(z_i)}{\sqrt{R}}$ instead of $\hat{\epsilon}_k(z_i)$ in equation (1.20), where $\epsilon_k(z_i|a,b,c,q,\xi)$ is given by equation (1.19), parameters for upper and lower confidence curves are estimated. These upper and lower confidence parameters can be used in equation (1.19) for ACER function prediction confidence intervals.

HUSK SKRIV OM PLOT FOR ACER TIL z I STEDET FOR η!!!!!!!!!!

1.2 Bayesian Inference

Change all α to ϕ for mcmc

For the traditional frequentist statistics, the parameters $\boldsymbol{\theta} = [\theta_1, \dots, \theta_m]$ are assumed fixed, while observations $\boldsymbol{x} = [x_1, \dots, x_n]$ are random from the underlying distribution $f(\boldsymbol{x}|\boldsymbol{\theta})$. Bayesian statistics, instead treats the parameters $\boldsymbol{\theta}$ with a probability distribution, where it is possible to make subjective believes about the distribution, independent of the data. These subjective beliefs is used to construct a prior distribution $f(\boldsymbol{\theta})$ based on experience, information or physical knowledge of the situation analyzed.

The posterior distribution of the parameters, dependent on the observed data becomes

$$f(\boldsymbol{\theta}|\boldsymbol{x}) = \frac{f(\boldsymbol{\theta})f(\boldsymbol{x}|\boldsymbol{\theta})}{\int_\Theta f(\boldsymbol{\theta})f(\boldsymbol{x}|\boldsymbol{\theta})\mathrm{d}\theta}, \tag{1.24}$$

where Θ is the domain over all possible parameters, for which the integral is taken, and $f(\boldsymbol{x}|\boldsymbol{\theta})$ is the likelihood function. The likelihood function is constructed from the joint density function, which for independent data equals

$$f(\boldsymbol{x}|\boldsymbol{\theta}) = \prod_{i=1}^{n} f(x_i|\boldsymbol{\theta}). \tag{1.25}$$

The integral over parameters reduces to a constant, which makes

$$f(\boldsymbol{\theta}|\boldsymbol{x}) \sim c \cdot f(\boldsymbol{\theta})f(\boldsymbol{x}|\boldsymbol{\theta}), \tag{1.26}$$

where $c = 1/\int_\Theta f(\boldsymbol{\theta})f(\boldsymbol{x}|\boldsymbol{\theta})\mathrm{d}\theta$ is the normalizing constant.

A conjugate prior is a prior which combined with the likelihood function construct a posterior distribution in the same family as the prior. Conjugate priors are often preferred because of the analytical luxury and computational simplicity.

Estimating a future point z which follows the distribution $f(z|\boldsymbol{\theta})$, can then be don using the posterior distribution $f(\boldsymbol{\theta}|\boldsymbol{x})$. The predicted point then becomes dependent of the observed data

$$f(z|\boldsymbol{y}) = \int_\Theta f(z|\boldsymbol{\theta})f(\boldsymbol{\theta}|\boldsymbol{x})\mathrm{d}\theta. \tag{1.27}$$

Since Bayesian inference accounts for the distribution of parameters, equation (1.13) can be rewritten using $\Pr(Z > u) = \psi$

$$\Pr(Z > z|\xi,\sigma,\psi) = \psi\left[1+\xi\left(\frac{z-u}{\sigma}\right)\right]^{-\frac{1}{\xi}}, \tag{1.28}$$

where ψ, ξ and σ are the unknown parameters. Since ψ is the probability of a point being larger than the threshold, ψ is independent of ξ and σ. Development of posterior distributions for independent parameters can be treated separately.

Starting with ξ and σ, by combining equation (1.25) and (1.10), the joint density function for the POT method becomes

$$\begin{aligned} f(\boldsymbol{y}|\xi,\sigma) &= \prod_{i=1}^{n} h(y_i|\xi,\sigma) \\ &= \sigma^{-n}\prod_{i=1}^{n}\left(1+\frac{\xi y_i}{\sigma}\right)^{-\left(1+\frac{1}{\xi}\right)}, \end{aligned} \tag{1.29}$$

or by (1.11) for $\xi = 0$, $f(\boldsymbol{y}|\sigma) = \sigma^{-n}\exp\left\{-\sigma^{-1}\sum_{i=1}^{n} y_i\right\}$. Here h is the probability density function of the GPD, $\boldsymbol{y}$ of threshold excess and n is the numbers of threshold excess.

The obvious beginning for investigating priors is the conjugate priors, but unfortunately, there do not appear to be any conjugate priors for the joint GPD. This work, will not go into depth on how to select Bayesian priors for the GPD, but instead use the suggestion proposed by (Coles, 2001, p. 174). Note that there are potential improvements by deeper investigation of GPD or GEV priors, especially for priors developed for specific situations where there are physical knowledge or practical experiences about the parameters. Since $\sigma > 0$, the transformation $\phi = \log(\sigma)$ ensures σ to be valid without restriction on ϕ. The suggested priors are $f_\phi(\cdot)$ and $f_\xi(\cdot)$ to be normally distributed around zero with variance $v_\phi = 10^4$ and $v_\xi = 100$.

Considering the prior distribution of ϕ instead of σ, the change of variable for the joint density function becomes

$$\begin{aligned} f_{\boldsymbol{y}|\xi,\phi}(\boldsymbol{y}|\xi,\phi) &= \frac{f_{\boldsymbol{y},\xi,\phi}(\boldsymbol{y},\xi,\phi)}{f_{\xi,\phi}(\xi,\phi)} \\ &= \frac{f_{\boldsymbol{y},\xi,\sigma}\left(\boldsymbol{y},\xi,\exp(\phi)\right)\cdot\left|\frac{\mathrm{d}}{\mathrm{d}\phi}\exp(\phi)\right|}{f_{\xi,\sigma}\left(\xi,\exp(\phi)\right)\cdot\left|\frac{\mathrm{d}}{\mathrm{d}\phi}\exp(\phi)\right|} \\ &= f_{\boldsymbol{y}|\xi,\sigma}\left(\boldsymbol{y}|\xi,\exp(\phi)\right), \end{aligned} \tag{1.30}$$

where $f_X(\cdot)$ indicates the probability distribution of X. The posterior distribution of the parameters can then be developed by the priors, (1.30) and (1.26)

$$f_{\xi,\phi|\boldsymbol{y}}(\xi,\phi|\boldsymbol{y}) \sim c\cdot f_{\boldsymbol{y}|\xi,\sigma}\left(\boldsymbol{y}|\xi,\exp(\phi)\right) f_\xi(\xi) f_\phi(\phi), \tag{1.31}$$

where again c is the normalizing constant, $f_{\boldsymbol{y}|\xi,\sigma}$ is as in (1.29), $f_\xi(\xi) \sim N(0,100)$ and $f_\phi(\phi) \sim N(0,10^4)$, and $N(\mu,\sigma^2)$ indicates the normal distribution with mean μ and variance σ^2.

The development of ψ posterior distribution, can be started with investigating priors. Since ψ simply equals $\Pr(X > u)$, the range is limited between 0 and 1. For simplicity the prior is set proportional to the uniform distribution on the interval $(0,1)$, $f(\psi) \sim UNIF(0,1)$. It is noted that in reality the distribution of ψ is not flat. Low valued ψ is more probable then high, while the probability converges to zero for the endpoints. A well-tuned Beta distributed prior could account for this, and improve the result.

The joint density function can be created by the fact that ψ simply equals the probability that a random event exceeds the threshold. This can be expressed using the binominal distribution, where k_i indicates the numbers of points exceeding the threshold and N_i indicates the total numbers of point, each for a given period i. For a total m numbers of periods, the posterior distribution equals

$$\begin{aligned} f(\psi|k_1,\ldots,k_n,N_1,\ldots,N_n) &\sim f(\psi)\cdot\prod_{i=1}^{m} f(k_i|N_i,\psi) \\ &= \prod_{i=1}^{m}\binom{N_i}{k_i}\psi^{k_i}(1-\psi)^{N_i-k_i} \\ &\sim \psi^{\Sigma_{i=1}^m k_i}(1-\psi)^{\Sigma_{i=1}^m N_i-\Sigma_{i=1}^m k_i}. \end{aligned} \tag{1.32}$$

The resulting distribution is independent of period selection, the notation k and N can be used for the total numbers of exceedance and the total numbers of measurements respectively. It is noted that the distribution is proportional to the Beta distribution. Since there only exist one normalizing constant which satisfies the requirements for a probability distribution, the posterior distribution is not only proportional, but equal to the Beta distribution. Rewriting equation (1.32) gives

$$f(\psi|k,N) \sim BETA(k+1, N-k+1). \tag{1.33}$$

1.3 Markov Chain Monte Carlo

Markov Chain Monte Carlo (MCMC) is a powerful iterative method used to sample from probability distributions, which analytically or through other simulation methods, could be difficult and impracticable to sample from. The algorithm is constructed by converging the desired probability distribution to an irreducible and aperiodic Markov Chain with limiting distribution equal the target distribution. Independent of starting position the Markov Chain will then converge towards the desired probability distribution in the limit as the numbers of iterations goes to infinity. The first numbers of realizations from the Markov Chain until converging is called the burn-in period, and is discarded for further analyses. More information about burn-in can be found in (Givens and Hoeting, 2013, p.220). The remaining realizations, approximately follows the desired probability distribution, where the accuracy increase as the numbers of realizations increase. Monte Carlo method can then be used to calculate the quantity of interest like mean, expected value, future prediction, credible interval etc. For more in depth description of the MCMC method see the books of Gamerman and Lopes (2006) and (Givens and Hoeting, 2013, Chapter 7,8).

1.3.1 Gibbs Sampling

In situations where it is difficult to sample from the joint distribution, but applicable from the conditional distribution, Gibbs sampler is preferable. The theory behind Gibbs sampling was first proposed in Geman and Geman (1984). The principle of Gibbs sampling is to construct the Markov Chain, by repeatedly sample each parameter with the rest of the parameters as the condition. The Gibbs sampler start with an initial guess $\boldsymbol{X}^0 = [X_1^0, \cdots, X_n^0]$ for the parameters $\boldsymbol{X}$, then iteratively update each as follows

$$\begin{aligned} X_1^{t+1}|\cdot &\sim f(X_1|X_2^t, \dots, X_n^t), \\ X_2^{t+1}|\cdot &\sim f(X_2|X_1^{t+1}, X_3^t, \dots, X_n^t), \\ &\vdots \\ X_n^{t+1}|\cdot &\sim f(X_n|X_1^{t+1}, \dots, X_{n-1}^{t+1}). \end{aligned} \tag{1.34}$$

where t is the iteration number, f is probability function of the parameter and $|\cdot$ symbolize that the function is conditional on the rest and recent parameters. In some cases, it could be beneficial to sample some of the parameters in blocks, such as $(X_k, X_{k+1})|\cdot$ where $1 \leq k \leq n$. This form of Gibbs sampling is called blocking. The iterative process is repeated until enough realizations are generated for sufficient accuracy. More on Gibbs sampling can be found in (Gamerman and Lopes, 2006, p. 141) and (Givens and Hoeting, 2013, p. 209)

1.3.2 Metropolis–Hastings Algorithm

The Metropolis-Hastings algorithm was first proposed by Metropolis et al. (1953), and is another method for constructing a suitable Markov Chain. The algorithm is preferable for situations where a proportional distribution is simple to evaluate, while the target probability distribution is difficult. Bayesian inference (see chapter 1.2), often result in a distribution where the normalizing constant cannot analytically be calculated. While possible numerically, the normalizing constant often becomes computationally hard, which make it impracticable for iterative simulations. Using Metropolis–Hastings algorithm on a proportional distribution whiteout normalizing constant, results in samples from the target distributions.

The Metropolis–Hastings algorithm start with an initial guess for the parameters. For each iteration a new parameters $\boldsymbol{X}^*$ are suggested from a proposal distribution $g(\boldsymbol{X}^*|\boldsymbol{X}^t)$, given the last accepted parameter $\boldsymbol{X}^t$. The new parameter is then evaluated against the last accepted by

$$R(\boldsymbol{X}^*, \boldsymbol{X}^t) = \frac{f(\boldsymbol{X}^*)g(\boldsymbol{X}^t|\boldsymbol{X}^*)}{f(\boldsymbol{X}^t)g(\boldsymbol{X}^*|\boldsymbol{X}^t)} \tag{1.35}$$

where $f(x)$ is the target distribution, or a distribution proportional to the target distribution. The parameter $\boldsymbol{X}^{t+1}$ takes value $\boldsymbol{X}^*$ with probability $\min\{1, R(\boldsymbol{X}^*, \boldsymbol{X}^t)\}$, if rejected, $\boldsymbol{X}^{t+1} = \boldsymbol{X}^t$ instead. The reason the target distribution normalizing constant is irrelevant, is because they are both canceled out in $f(\boldsymbol{X}^*)/f(\boldsymbol{X}^t)$.

A common proposal distribution is the random walk. The new parameters are generated from the last accepted realization with additional variance, $\boldsymbol{X}^* = \boldsymbol{X}^t + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon}$ follows a chosen probability distribution. Symmetric Proposals implies that $g(\boldsymbol{X}^t|\boldsymbol{X}^*) = g(\boldsymbol{X}^*|\boldsymbol{X}^t)$, this is referred to as Metropolis algorithm.

The Metropolis–Hastings algorithm could in situations be necessary for some of the steps in the Gibbs sampler, equation (1.34). Such a combination of Metropolis-Hastings algorithm and Gibbs sampler is referred to as a Hybrid Gibbs sampler, and was first introduced by Müller (1991). For more information about the Metropolis-Hastings algorithm and Hybrid Gibbs sampler see (Givens and Hoeting, 2013, p. 202), (Gamerman and Lopes, 2006, p. 191) and (Givens and Hoeting, 2013, p. 216), cite[p. 205]MCMC respectively.

1.3.3 Effective Sample Size

The realizations of the simulated Markov Chain will often be correlated, and dependent on the future and past iterations. The correlation implies that the information hold by the realizations is actually less than the numbers of realizations. The effective sample size gives a method of calculating the theoretical size of an equally informative i.i.d., realization set. The effective sample size is estimated as

$$L_{eff} = \frac{L}{1 + 2\sum_{k=1}^{K} \hat{\rho}(k)}, \tag{1.36}$$

where L is the sample size of the simulated realizations, $\hat{\rho}(k)$ is the estimated k step autocorrelation between realizations and K is chosen as the first k where $\hat{\rho}(k) < 0.1$. The effective sample size is a quantification of the information hold by the simulated realization set.

1.3.4 Adaptive Metropolis Algorithm

A challenge with constructing a MCMC is to ensure that the series converge to the stationary target distribution relatively quickly, and that the samples gives points in the whole range of the target distribution, this is referred to as good mixing.

If a large percentage of Metropolis-Hastings proposal $\boldsymbol{X}^*$ is accepted, the proposal distribution is too narrow. High acceptance rate will delay convergence, and cause higher correlation between points. The result is poor mixing and a decrease in effective sample size. On the other hand, if only a small percentage of the proposals is accepted, the proposal distribution is to wide. Low acceptance rate will also increase correlation, which gives poor mixing and decreased effective sample size. A large number of generated realizations by the Markov chain will be equal, which will harm future Monte Carlo simulation.

To maximize the effective sample size and ensure good mixing, the acceptance rate should be somewhere in between. For a Metropolis-Hastings algorithm, Gelman et al. (1996) suggested a 44% acceptance rate for single dimensional normal target distribution and 23.4% for high dimensional multivariate normal target distribution. Commonly the user would run the Metropolis-Hastings algorithm, calculate acceptance rate, tune variance and then rerun the process until sufficient acceptance rate is achieved.

For this work, MCMC simulation will be used for a large number of different situations, and it would become extremely time-consuming to tune each variance. The inconvenient can be handled by using an Adaptive Markov Chain Monte Carlo (AMCMC) which adapt the MCMC algorithm while running. This is achievable using a normal random walk proposal where the next suggested realization $\boldsymbol{X}^* \sim N(\boldsymbol{X}^t, \lambda\boldsymbol{\Sigma}^t)$. Between iterations $\boldsymbol{\Sigma}^t$ is adjusted to improving mixing and efficient sample size. The acceptance rate is set by λ, and with a p dimensional multivariate normal target distribution, it has been shown that a constant $\lambda = 2.38^2/p$ is optimal when $\boldsymbol{\Sigma}$ equals the real variance of the target distribution, Gelman et al. (1996). The adaptive Metropolis algorithm is not constraint to the normal target distributions, but the suggested λ seems like a good starting value. The ability of an adjustable λ between future iteration seems beneficial, because of the unknown target distribution and the following acceptance rate. The additional adaptive parameter $\boldsymbol{\mu}^t$ is necessary since the covariance is proportional to $\boldsymbol{\mu}$. The initial guess is chosen as $\boldsymbol{\mu}^0 = \mathbf{0}$ and $\boldsymbol{\Sigma}^0 = \mathbf{I}$. The normal random walk proposal distribution is symmetric, which result in an adaptive Metropolis algorithm, where (1.35) is reduced to

$$R(\boldsymbol{X}^*, \boldsymbol{X}^t) = \frac{f(\boldsymbol{X}^*)}{f(\boldsymbol{X}^t)}. \tag{1.37}$$

For each iteration $\boldsymbol{\mu}^{t+1}$ and $\boldsymbol{\Sigma}^{t+1}$ is updated as follows

$$\boldsymbol{\mu}^{t+1} = \boldsymbol{\mu}^{t} + \gamma^{t+1}(\boldsymbol{X}^{t+1} - \boldsymbol{\mu}^{t}) \tag{1.38}$$

$$\boldsymbol{\Sigma}^{t+1} = \boldsymbol{\Sigma}^{t} + \gamma^{t+1}\left[(\boldsymbol{X}^{t+1} - \boldsymbol{\mu}^{t})(\boldsymbol{X}^{t+1} - \boldsymbol{\mu}^{t})^{T} - \boldsymbol{\Sigma}^{t}\right], \tag{1.39}$$

Where γ is a decreasing parameters which provide the Markov chain property described in the beginning of chapter 1.3. The details of γ^t, to ensure an irreducible and aperiodic Markov chain can be found in Roberts and Rosenthal (2007) and Atchadé et al. (2011) , where it is noted that $\lim_{t->\infty}\gamma^t = 0$ while the summation, not necessary is bounded $\sum_{t=1}^{\infty}\gamma = \infty$. Repeated trails concluded that $\gamma^t = \cdots$ HUSK AA FYLL INN HER!!!!!!!!!! was sufficient choice.

As described above an adaptive λ^t could be beneficial. By using

$$\log(\lambda^{t+1}) = \log(\lambda^{t}) + \gamma^{t+1}\left(R(\boldsymbol{X}^{*}, \boldsymbol{X}^{t}) - a\right), \tag{1.40}$$

the series acceptance rate will converge towards a (Givens and Hoeting, 2013, p. 248).

A more detailed description of the adaptive metropolis algorithm can be found in (Givens and Hoeting, 2013, p. 247). CHECK BOLDSYMBOL FOR VECTOR AND MATRIX!!

1.3.5 Applying Markov Chain Monte Carlo to Peak Over Threshold

From chapter 1.2, the two equation (1.31) and (1.33) construct the basis for the blocking Gibbs sampler

$$\xi^{t+1}, \phi^{t+1}|\cdot \sim c \cdot f_{\boldsymbol{y}|\xi,\sigma}\left(\boldsymbol{y}|\xi, \exp(\phi)\right) f_{\xi}(\xi) f_{\phi}(\phi)$$

$$\psi^{t+1}|\cdot \sim BETA(k+1, N-k+1),$$

where the parameters and functions are described above in chapter 1.2. After the Markov chain sampling is complete, the transformation $\sigma = \exp(\phi)$ ensure correct parameter for the Monte Carlo simulation's. Sampling form ψ is straight forward since it is simply realizations of the Beta distribution, while ξ, σ is more complex, and cannot directly be sampled. The challenge of calculating the computationally heavy c for each iteration favors the implementation of the

Metropolish-Hastings algorithm.

The algorithm independency of the user for tuning and improved convergence speed makes the adaptive Metropolis-Hasting algorithm, described in chapter 1.3.4, favorable for ξ, σ. The posterior distribution often results in extremely small values, which in some cases could get disrupted by the violation of the smallest floating number for the software. To account for this, the logarithm of equation (1.37) is used. The resulting log Metropolis ratio becomes,

$$\begin{aligned}\ln\left[R(\boldsymbol{X}^*, \boldsymbol{X}^t)\right] &= \ln\left[f_{\xi,\phi|\boldsymbol{y}}(\xi^*, \phi^*|\boldsymbol{y})\right] - \ln\left[f_{\xi,\phi|\boldsymbol{y}}(\xi^t, \phi^t|\boldsymbol{y})\right] \\ &= \ln\left[f_{\boldsymbol{y}|\xi,\sigma}\left(\boldsymbol{y}|\xi^*, \exp(\phi^*)\right)\right] - \ln\left[f_{\boldsymbol{y}|\xi,\sigma}\left(\boldsymbol{y}|\xi^t, \exp(\phi^t)\right)\right] + \\ &\quad \ln\left[f_\xi(\xi^*)\right] + \ln\left[f_\phi(\phi^*)\right] - \ln\left[f_\xi(\xi^t)\right] - \ln\left[f_\phi(\phi^t)\right],\end{aligned} \tag{1.41}$$

where equation (1.29) gives,

$$\ln\left[f_{\boldsymbol{y}|\xi,\sigma}\left(\boldsymbol{y}|\xi, \exp(\phi)\right)\right] = \begin{cases} -n\phi - \exp(-\phi)\sum_{i=1}^n y_i, & \xi = 0, \\ -n\phi - \left(1 + \frac{1}{\xi}\right)\sum_{i=1}^n \ln\left(1 + \xi\exp(-\phi)y_i\right), & \text{else}, \end{cases} \tag{1.42}$$

and after inserting the priors mean and variance from chapter 1.2, the remaining parts reduces to

$$\ln\left[f_\xi(\xi^*)\right] + \ln\left[f_\phi(\phi^*)\right] - \ln\left[f_\xi(\xi^t)\right] - \ln\left[f_\phi(\phi^t)\right] = -\frac{(\xi^*)^2 - (\xi^t)^2}{200} - \frac{(\phi^*)^2 - (\phi^t)^2}{2 \cdot 10^4}. \tag{1.43}$$

For the equations, t indicate the parameter iteration number, * indicate the proposed parameter value to be evaluated, ξ and σ are GPD parameters where $\phi = \log(\sigma)$ and $\boldsymbol{y}$ is a vector containing the observed data of size n.

The remaining construction of the AMCMC simmulator is as described above in chapter 1.3.4. The result is a hybrid Gibbs AMCMC simulator, for the POT method. The Markov chain is valid since both Gibbs steps are irreducible and aperiodic. Irreducible because each sampler within their restricted range can sample any realization with probability larger than zero, from any state. Aperiodic since both Gibbs steps can return to their state in a single iteration, with probability larger than zero.

To limit the possibility of the adaptive Metropolis-Hastings algorithm getting stuck in a slowly converging area before reaching the target distribution, multiple independent MCMC simulations with different starting values for ξ^0 and ϕ^0 are used. After convergence the latest value of ξ^t and ϕ^t between the MCMC simulations with the highest $f_{\xi,\phi}(\xi^t,\phi^t)f(\xi^t)f(phi^t)$ is selected for future realization generation, while all points generated prior to this point for each starting value is discarded as burn-in.

Estimation and credible interval can simply be added through Monte Carlo simulation. For a parameter realizations of size T after burn-in is removed, where the effective sample size is sufficient, equation (1.27) is estimated as

$$\hat{\Pr}(Z > z|\boldsymbol{y}) = \frac{1}{T}\sum_{t=1}^{T}\Pr(Z > z|\xi^t,\sigma^t,\psi^t). \tag{1.44}$$

The estimation is unbiased since $(\xi^t,\sigma^t,\psi^t) \sim f(\xi,\sigma,\psi|\boldsymbol{y})$. The $100\cdot\alpha\%$ credible interval for a specific $z = s$ can be added to the $\hat{\Pr}(Z > s|\boldsymbol{y})$ estimation by sorting the result of $\Pr(Z > s|\xi^t,\sigma^t,\psi^t)$ for every realizations, and selecting lower and upper limits for where $100\cdot\alpha\%$ of the results are contained. The narrowest interval is chosen for this work and is called the highest posterior density interval. Example of the use of such interval can be seen in figure 3.3, where the credible interval has been estimated on multiple z values for the MCMC method.

1.3.6 Multivariate Random Normal Generator

Most of the coding for this work was done in R. Since the MCMC accuracy increase with the numbers of iteration generated, parts of the AMCMC algorithm was coded in C++, through the Rcpp package by Eddelbuettel et al. (2016), for speed optimization. The AMCMC for ξ and ϕ uses a bivariate random normal generator, but this is not natively supported in C++. Since no additional packages tested was satisfactory for the purpose, the bivariate random normal generator was constructed.

The Box-Muller transformation, see Box and Muller (1958), states that for two independent random variables $U_1, U_2 \sim UNIF(0,1)$, the transformation

$$z_1 = \sqrt{-2\ln(U_1)}\,\cos(2\pi U_2)$$
$$z_2 = \sqrt{-2\ln(U_1)}\,\sin(2\pi U_2)$$

results in Z_1 and Z_2 to be independent and standard normally distributed. Combined in a vector $\boldsymbol{z} = [Z_1, Z_2]^T$ where T is the transpose, the bivariate random normal $\boldsymbol{x} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ can be generated by

$$\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{A}\boldsymbol{z}, \tag{1.45}$$

where $\boldsymbol{\Sigma} = \boldsymbol{A}\boldsymbol{A}^T$. For this work $\boldsymbol{A}$ is chosen as the Cholesky decomposition of $\boldsymbol{\Sigma}$.

STATE SOMEWHERE THAT log IS THE NATURAL LOGRATIM?

1.4 Forecasting Extreme

For this work two methods were used for prediction and forecasting of future extreme. IT IS CALLED PREDICTION INTERVAL NOT CONFIDENCE/CREDIBLE INTERVAL!! CONFIDENCE/-CREDIBLE INTERVALS ARE FOR MEAN

1.4.1 Value at Risk

Value at risk (VaR) for confidence α is defined as the smallest value l which the loss L of the portfolio will exceed with probability α, over a given time period h

$$\Pr(L_h \geq l_{h,\alpha}) = \alpha. \tag{1.46}$$

As this work only considers 1 day VaR the simplified notation $\mathrm{VaR}_\alpha = l_\alpha$ is used.

By inverting equation (1.19), VaR of the ACER method for a given k is calculated as

$$\mathrm{VaR}_\alpha = b + \left[\frac{1}{a\xi}\left(\frac{q}{\alpha}\right)^{\xi} - 1\right]^{1/c}. \tag{1.47}$$

For the POT MCMC method, equation (1.28) is inverted, for which the VaR follows

$$\mathrm{VaR}_\alpha^t = u + \frac{\sigma^t}{\xi^t}\left[\left(\frac{\psi^t}{\alpha}\right)^{\xi^t} - 1\right], \tag{1.48}$$

where t indicates the sampled realization number from the MCMC. As the reasoning for equation (1.44), the estimated VaR becomes $\mathrm{VaR}_\alpha = 1/T\sum_{t=1}^{T}\mathrm{VaR}_\alpha^t$, where T is the total numbers of realizations generated after burn-in is removed.

Dependent if the portfolio is on buy or sell, the value at risk make sense for both increase and decrease of daily return respectively. For VaR on decreasing data, the extreme of the negative dataset $-X_1,\dots,-X_n$ is analyzed instead. IS THIS ACTUALLY DONE FOR DECREASING DAILY RETURNS?

1.4.2 Forecasting Extreme Confidence Interval

The VaR gives an approach for testing how well the ACER and POT MCMC methods estimates the underlying distribution. The VaR estimation does not reflect how well the ACER and POT MCMC captures the estimation variability. By analyzing the maximum of a future time series, a method for testing variability is developed.

As stated earlier in chapter 1.1.1, for time series with no trend, the most extreme events are approximately i.i.d, thus at the tails, equation (1.2) can be approximated as (1.3). The estimated cumulative distribution $\hat{F}(z)$ can thereby be used for estimating the maximum of a future long time series by

$$\Pr(M_n < z) \approx \left[\hat{F}(z)\right]^n, \tag{1.49}$$

where n is the number of points in the time series, as in chapter 1.1. Both the ACER and POT MCMC method is developed for $\hat{\Pr}(X > z) = 1 - \hat{F}(z)$, see equation (1.19) and (1.28) respectively. The $100 \cdot \alpha\%$ confidence interval of the most extreme of n points, or nth extreme, can be found by

$$\alpha = \Pr(m_{n,p_1} \leq M_n \leq m_{n,p_2}), \tag{1.50}$$

where $\Pr(M_n < m_{n,p_1}) = p_1$, $\Pr(M_n < m_{n,p_2}) = p_2$, $p_2 - p_1 = \alpha$, and both p_1 and p_2 is between 0 and 1. Using equation (1.49), the value of $m_{n,p}$ for a given n and p can be estimated by inverting (1.19), resulting in

$$\hat{m}_{n,p} = b + \left[\frac{1}{a\xi}\left(\frac{q}{1-p^{1/n}}\right)^{\xi} - 1\right]^{1/c}. \tag{1.51}$$

for the ACER method. By computing $\hat{m}_{n,p_1}$ and $\hat{m}_{n,p_2}$ for a variety of p_1, p_2 satisfying $\alpha = p_2 - p_1$, the highest posterior density interval is chosen as confidence interval.

The credible interval for the POT MCMC method is calculated by generating realizations of the nth extreme z_n, using various versions of parameter realizations generated by the MCMC method. Values of the estimated z_n can be generated using the probability integral transform approach, where $\Pr(X < x) \sim UNIF(0,1)$, combined with equation (1.28) and (1.49), this gives

$$\hat{z}_n^i = u + \frac{\sigma^t}{\xi^t}\left[\left(\frac{\psi^t}{1 - y_i^{1/n}}\right)^{\xi^t} - 1\right], \tag{1.52}$$

where $y_i \sim UNIF(0,1)$, i is the ith realization of $\hat{z}_n$ and t is the parameter realization number. For a high number of realization, the $100 \cdot \alpha\%$ highest posterior density interval is chosen by sorting $\hat{z}_n^i$ by values, and selecting the narrowest continues interval which holds $100 \cdot \alpha\%$ of the data. The POT MCMC credible interval also captures the parameter variation, unfortunately this is not the case for the confidence interval generated by the ACER method and therefor will result in a too narrow interval.

By definition the median of a cumulative distribution function is the number a that satisfy $\Pr(X < a) = 1/2$, consequently the median of the nth extreme equals $m_{n,1/2}$. The most extreme of n points can also be expressed by a probability $1/n$, for example the 0.01 extreme equals the 1 in 100 or $n = 100$ extreme. UNIF, BETA osv, uset text. "By definition the median" kanskje ikke noedvendig??

1.5 Test and Evaluation

1.5.1 Likelihood-ratio test

Likelihood ratio test is a method of comparing different models goodness of fit. The test required a null model $\boldsymbol{\theta_0}$, and an alternative model $\boldsymbol{\theta_a}$, where the null model is a nested subset of the alternative. The test statistics is approximately chi-square distributed, with $df_a - df_0$ degrees of freedom, where df_x is the models x number of free parameters. The log likelihood ratio test statistics have the following relationship

$$\chi^2_{(df_a - df_0)} \sim 2\left[\log\left(f(\boldsymbol{x}|\boldsymbol{\theta_a})\right) - \log\left(f(\boldsymbol{x}|\boldsymbol{\theta_0})\right)\right], \tag{1.53}$$

where $f(\boldsymbol{x}|\boldsymbol{\theta})$ is the likelihood function seen in equation (1.25) and χ^2 is the chi-square distribution.

1.5.2 AIC and BIC

mention AIC and BIC best is minimum AIC value lowest BIC is preferred

1.5.3 Evaluating Forecasts

blablabla likelihood ratio, Kupiec (1995), Christoffersen (1998).

And scoring table (how to).

1.6 ARMA-APARCH

The Autoregressive Moving Average (ARMA) model, is a method of analyzing time series. The ARMA(p,q) model for a time series $Z_1,\ldots,Z_T$, where $\dot{Z}_t = Z_t - \mu$ and μ is the mean of $\boldsymbol{Z}$, is given by

$$\dot{Z}_t = \sum_{i=1}^{p} \phi_i \dot{Z}_{t-i} \sum_{j=1}^{q} \theta_j e_{t-j} + e_t, \tag{1.54}$$

where ϕ is the Autoregressive (AR) parameters, θ is the Moving Average (MA) parameters and e is the error term. For $q=0$ and $p=0$ the ARMA(p,q) model is referred to as AR(p) and MA(q) respectively.

In situations where the error term is heteroscedastic, an Asymmetric Power Autoregressive Conditional Heteroscedasticity (APARCH) model can be included. For the time series above, the APARCH(k,l) model is as follows

$$\sigma_t^\delta = \omega + \sum_{k=1}^{r} \alpha_k \left(|e_{t-k}| - \gamma_k e_{t-k}\right)^\delta + \sum_{l=1}^{s} \beta_l (\sigma_{t-l})^\delta, \tag{1.55}$$

where $e_t = \sigma_t \epsilon_t$ and ϵ_t is a homoscedastic error term. The APARCH model equals the Autoregressive Conditional Heteroscedasticity (ARCH) for $\delta = 2$, $\gamma = 0$, $\beta = 0$, the Generalized Autoregressive Conditional Heteroscedasticity (GARCH) for $\delta = 2$, $\gamma = 0$ and the Glosten Jagannathan Runkle (GJR) GARCH for $\delta = 2$. The APARCH model has shown to work well for fat tails, excess kurtosis and leverage effects. For the data analyzed in this work, the error term ϵ_t is assumed to follow a skewed student t distribution. Thus $\epsilon_t \sim \text{Skew}(\mu = 0, \sigma = 1, \nu, \xi)$, where μ is location, σ is scale, ν is shape and ξ is the skewness parameters. The theory behind the skewed student t distribution is not presented here, but a detail description can be found in Fernandez and Steel (1998).

Chapter 2

Data

The following chapter introduce the different types of data analyzed in this work. Synthetic data are generated an analyzed, for better control of behavior and result verification of the ACER and POT MCMC method before the methods are used for a variety of commodity times series.

2.1 Synthetic Data

Two synthetic data series are generated. Both attempts to test the methods on different aspects which are important for real life commodity analysis. The first is the thick tail Pareto distribution, since commodity data are fat tail distributed, as was concluded by Aloui and Mabrouk (2010). The Pareto distribution can generate i.i.d. data which are easy to control and where exact analytic inference can be achieved.

The second, numerically generates a time series with approximately the same distribution characteristics as the return of crude oil commodity data. Since it is a numeric method, the data can be generated as large as desired, and much larger than is practical achievable in real life. This makes test result and verification much more accurate than limited real life data.

2.1.1 Pareto Distribution

The Pareto distribution has the following cumulative distribution function

$$\Pr(X < x) = \begin{cases} 1 - \frac{1}{x^{\beta}} & x > 1 \\ 0 & x < 1 \end{cases} \tag{2.1}$$

where β is a shape parameter. A cumulative distribution function is said to have fat tail if $\Pr(X > x) \sim x^{-\beta}$, as $x \to \infty$, for $\beta > 0$. By the definition, the Pareto distribution clearly has a fat tail. Using the probability integral transform approach, a data point x_i is generated by

$$x_i = \frac{1}{u_i^{1/\beta}}, \tag{2.2}$$

where $u_i \sim UNIF(0,1)$.

A practical data size could come from having a daily return each day for 25 years. Without considering leap year, roll-over returns, and weekend, that corresponds in a 25 times 365 data series. Multiple data series with this size was generated using $1 < \beta < 5$. Figure 2.1 show a generated Pareto distributed time series for $\beta = 3$, where the realizations range from 1.000 to 23.384.

2.1.2 Generated Commodity Data

As the Pareto distribution above generated i.i.d. data points, it seems logical to simulate dependent and homoscedastic data to reveal any performance difference of the methods. This work focus on the ACER and POT MCMC methods ability to capture the tail effect of commodity data, hence the ability to generating data with close to the same behaviors as real life commodity is desired.

It was suggested by Giot and Laurent (2003) that The AR(3)-APARCH(1,1), with skewed student t distribution as error term, was a good approximation for the commodity time series. Starting with the suggested model, the AIC, BIC and log likelihood ratio test was applied to the neighboring models, for model selection. Using the observed crude oil daily return described below in chapter 2.2, the parameters for AR(3)-APARCH(1,1), and neighboring models, can be

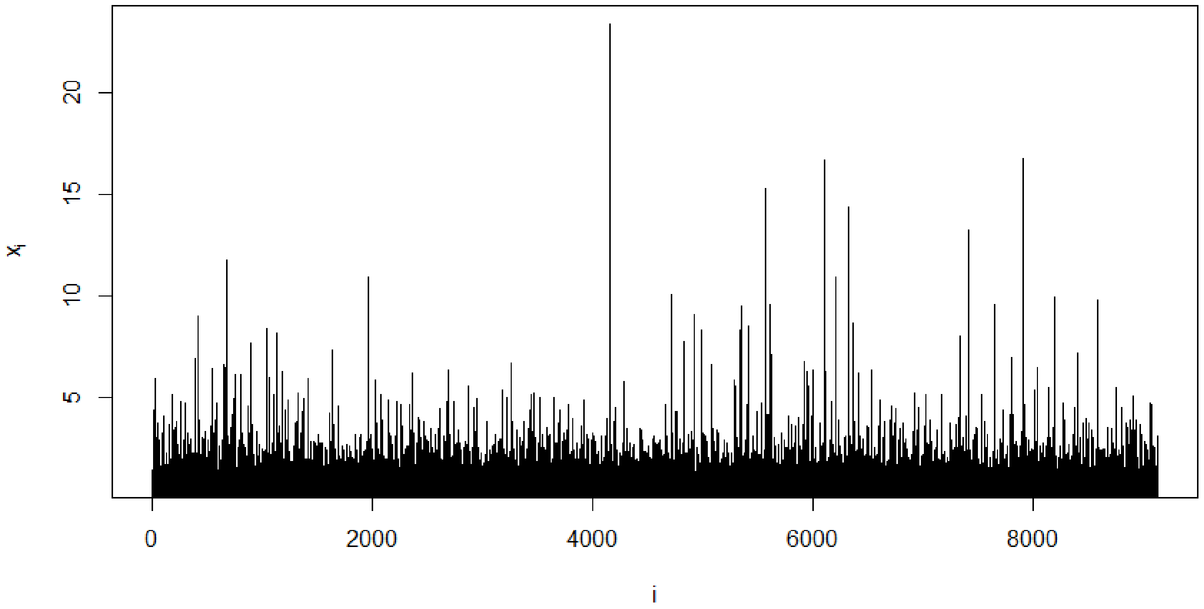

Figure 2.1: Generated Pareto distributed points with 9125 realization and $\beta = 3$. The i axis represent time in days, while x_i is the realized data point for that day.

estimated. The AIC, BIC and the log likelihood ratio test favors a higher AR dependency, while MA and additional APARCH parameters does not improve the model significantly. As the test show sign of a high AR dependency, selecting the model comes down to accuracy against computational intensity. The AR(4)- APARCH$(1,1)$ is concluded as a reasonable good model with lower AIC and BIC than its neighbors, and the log likelihood ratio test showing significantly improvements from all subsets, while AR(5)- APARCH$(1,1)$ does not significantly improve the model. The estimated parameters of AR(4)- APARCH$(1,1)$ with $\mu = 6.34 \cdot 10^{-4}$ is presented in table 2.1.

Giot and Laurent (2003)

2.2 Commodity Data

Adjusted for the roll-over returns. (simply done by deleting the return at the roll-over date).(roll-over, en kontrakt til neste eks sommer til hoest, kan gi ekstreme returns som gir forvrengende

	Estimate	p-value
ϕ_1	$-2.586 \cdot 10^{-2}$	0.0833
ϕ_2	$-4.339 \cdot 10^{-2}$	$3.66 \cdot 10^{-3}$
ϕ_3	$-1.699 \cdot 10^{-2}$	0.236
ϕ_4	$-4.682 \cdot 10^{-3}$	0.751
ω	$9.869 \cdot 10^{-5}$	$2.43 \cdot 10^{-3}$
α_1	$5.557 \cdot 10^{-2}$	$< 2 \cdot 10^{-16}$
γ_1	$2.215 \cdot 10^{-1}$	$356 \cdot 10^{-3}$
β_1	$9.498 \cdot 10^{-1}$	$< 2 \cdot 10^{-16}$
δ	1.123	$1.71 \cdot 10^{-7}$
ξ	$9.682 \cdot 10^{-1}$	$< 2 \cdot 10^{-16}$
ν	7.333	$< 2 \cdot 10^{-16}$

Table 2.1:

resultater). blabla

Chapter 3

Analysis and Results

The following chapter contains the analysis of the data presented in chapter 2. First is the analysis of the two computer generated synthetic data, before the methods are applied to the real life commodity data.

For the POT MCMC method, as described in chapter 1.3.4, the optimal acceptance rate for a single dimensional normal target distribution is 44% while it is 23.4% for higher dimensional multivariate normal target distribution. The parameters ξ and ϕ probably does not follow a bivariate normal target distribution, and they are only two dimensional, hence the optimal acceptance rate is unknown. Calculating the effective sample size for multiple different MCMC simulations on different data sets, using a variety of acceptance rate between 20% and 50%, shows a trend suggesting $30\% \leq a \leq 40\%$. For the rest of this work a is set to 0.35, which will converge the AMCMC towards an acceptance rate of 35%, by the theory described in chapter 1.3.4.

3.1 Analysis of Syntetic data

The goal of this section is to use the ACER and POT MCMC method to analyze controlled data for better conformation of prediction and test reliability. The Pareto distribution is i.i.d. while the simulated AR(1)-APARCH(1,1)?, is not. It is also interesting to study the performance difference for the two method in each cases.

3.1.1 Pareto Distirbution

One of the advantage of analyzing the Pareto distributions, is that the theoretical distribution is known, hence the exact analytical values can be achieved even for extreme cases. For testing, the yearly, decadal, centennial and millennial event or in probability 1/365, 1/3650, 1/36500 and 1/3650000 respectively, are estimated and compared with the exact values.

Given the definition of VaR equation (1.46), the exact VaR for the Pareto distribution can be found by

$$\text{VaR}_\alpha = \frac{1}{\alpha^{1/\beta}}.$$

For the estimated prediction interval, the exact probability for a future nth maximum to arriving whiten that limit is developed by combining equation (1.3) and (2.1), which result in

$$\Pr\left(m_n^{(-)} \leq M_n \leq m_n^{(+)}\right) = \left(1 - m_n^{(+)^{-\beta}}\right)^n - \left(1 - m_n^{(-)^{-\beta}}\right)^n,$$

where $m_n^{(+)}$ and $m_n^{(-)}$ is the upper and lower limits respectively.

The Pareto data from figure 2.1, with $\beta = 3$ is analyzed for a walkthrough of how the methods are used, and result interpretation. The exact VaR for each case are represented in table 3.1.

Event	α	VaR_α
Yearly	1/365	7.147
Decadal	1/3650	15.397
Centennial	1/36500	33.171
Millennial	1/365000	71.466

Table 3.1: The exact VaR for each event with the corresponding probability α, for the Pareto distribution with $\beta = 3$.

Since the data are i.i.d., the ACER methods will use $k = 1$, and the POT MCMC method will not need any decluttering or threshold analysis. In theory the point for where regular tail behavior starts for the ACER method, and the threshold for the MCMC method could be selected to 1, which would make use of the entire data set without removing lower data. In real life such a situation is unrealistically, thus both is selected to 1.5. Consequently, out of the total 9 125 Pareto distributed points, 2 728 exceeds 1.5 and is used for future analysis.

The POT MCMC method is set to generate 100 000 realizations. In figure 3.1 the first 1 000 realizations is shown with starting conditions $[\xi^0, \phi^0] = [1, 1]$. The plot converges within a few

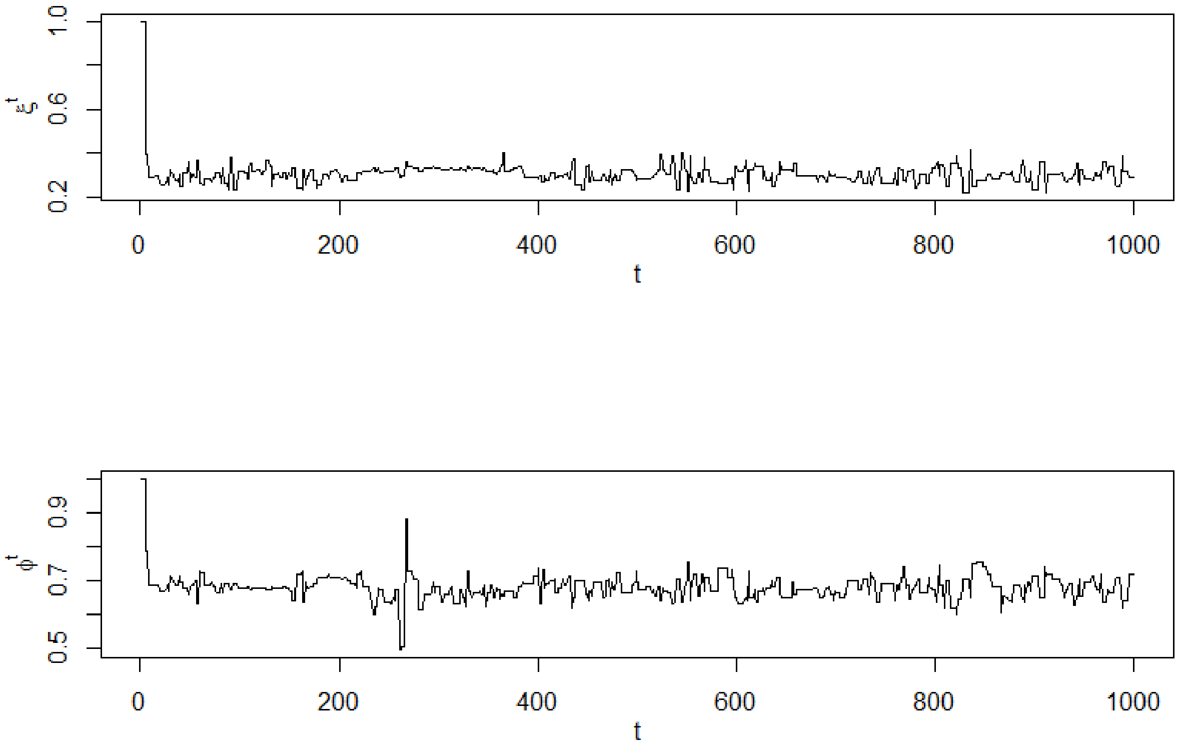

Figure 3.1: The first 1 000 realizations of the POT MCMC method for ξ and ϕ.

numbers of steps, while the variance of the points is adapting for some more iterations. Points generated after convergence, does follow the correct distribution, the variance only affects the effective sample size, even though it has not settled in yet. Data between 1st and 500th realization is selected as burn-in and is discarded for later analysis, resulting in an effective sample size of 13 580 for ξ and 14 746 for σ. The realized distribution of ξ, σ and β for the POT MCMC method can be seen in figure 3.2

For the ACER method, the estimates, together with the upper and lower confidence limits can be seen in table 3.2, with corresponding lines plotted in figure 3.3.

Comparison of the ACER and POT MCMC estimated probability functions with the real Pareto distribution can be seen in figure 3.3. Defining loss as an increase in value, the plot can also be interpreted as α against VaR_α. It is noted that the estimates for the POT MCMC are closer to the real distribution, with slimmer confidence interval. It is not surprisingly that the

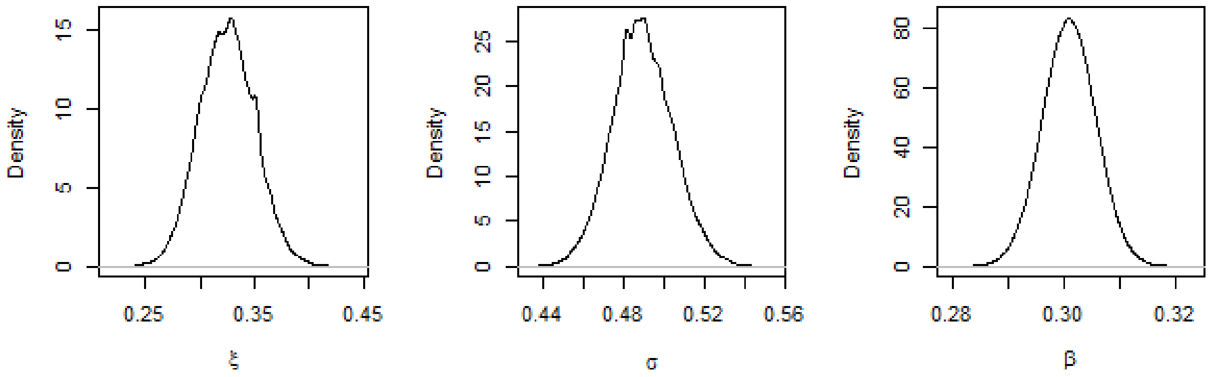

Figure 3.2: The estimated posterior density of the POT MCMC parameters ξ, σ and β.

	a	b	c	q	ξ
Upper 95% CI line	2.824	1.029	1.626	0.577	0.769
Estimated line	2.756	1.000	1.356	0.690	0.545
Lower 95% CI lien	3.105	1.000	0.652	1.747	0.110

Table 3.2: The estimated ACER parameters together with the upper and lower 95% confidence interval line parameters.

POT MCMC method perform better than the ACER method, since the points are i.i.d. The ACER methods capability of capturing data dependency is not necessary for the Pareto distribution, while the POT MCMC method uses an unrealistically large dataset for estimation.

Using the theory from chapter 1.4.2, the distribution of the maximum event for the upcoming year, decade, century and millennium is prediction for the two methods. Figure 3.4 shows the distribution of the exact Pareto prediction together with the estimated ACER and POT MCMC, with prediction intervals.

An approach for validating the methods prediction interval, can come from calculating how much of the real distribution was contained within the predicted intervals. See table 3.3 for the corresponding values. The table shows that both methods in each case predict reasonable

	year	decade	century	millennium
ACER	93.5% (13.9)	94.8% (34.8)	93.6% (87.58)	87.1% (220.9)
POT MCMC	92.0% (12.7)	92.8% (28.9)	94.1% (67.6)	95.4% (159.9)

Table 3.3: The amount of the exact distribution contained in the predicted 90% interval, with interval length in brackets, for the ACER and POT MCMC method.

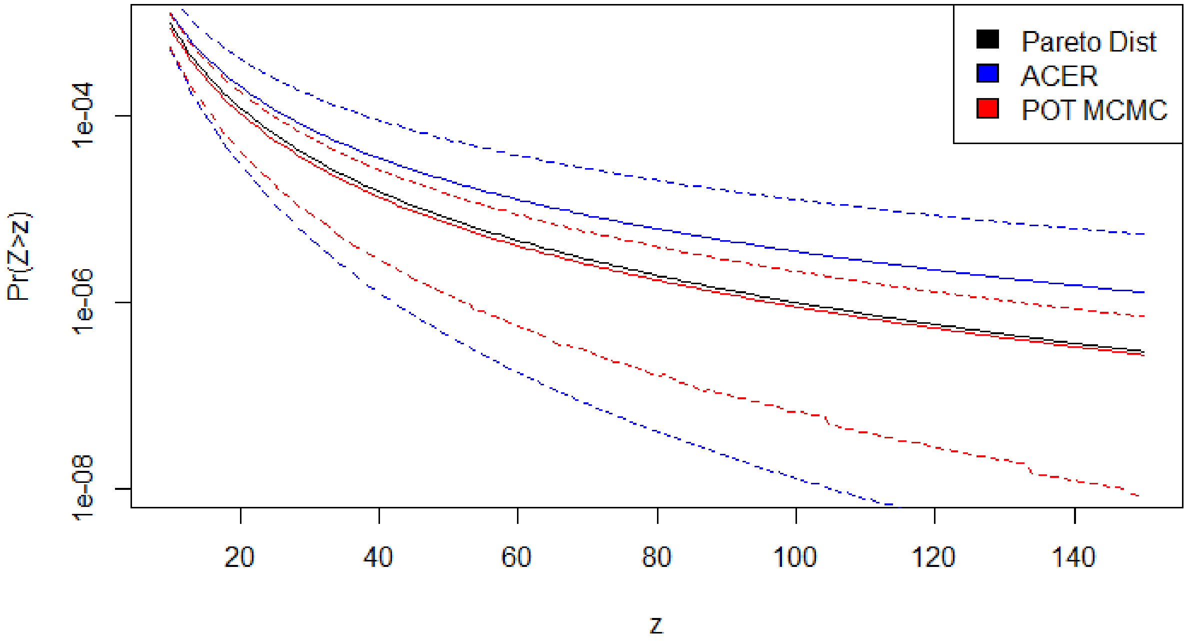

Figure 3.3: A plot visualizing the estimated ACER and POT MCMC probabilities, together with the exact Pareto distribution. The estimates are shown in solid lines while the upper and lower 95% confidence and credible limits are dashed.

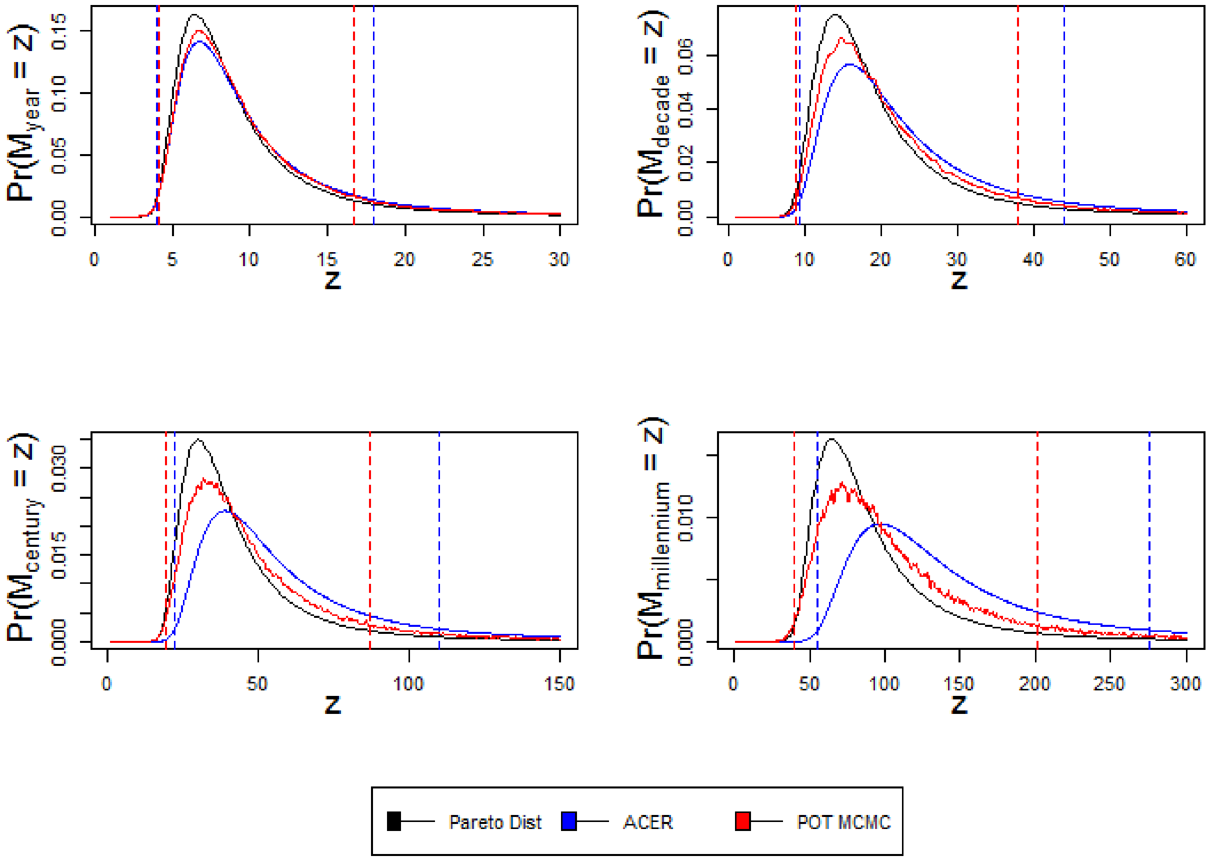

Figure 3.4: The estimated ACER and POT MCMC future year, decade, century and millennium predicted probability functions together with the true distribution. Solid line shows the probability distribution functions while dashed lines mark each methods corresponding 90% prediction interval.

close to 90% interval, while the POT MCMC method consistently perform better with respect to interval width.

By regenerating 30, Pareto distribution 9 125 points series, using a random β between 1 and 5 for each, the investigation above can be scaled up for a more accurate analysis.

DEFINER VARDIER SOM $\mu_{V\hat{a}R-VaR}$ OSV, KALK FOR MANGE, LAG TABELL SOM PAA TAVLEN!

3.1.2 Commodity Generated Data

3.2 Commodity Data

Appendix A

Additional Information

This is an example of an Appendix. You can write an Appendix in the same way as a chapter, with sections, subsections, and so on.

A.1 Introduction

A.1.1 R

```
#Dette er en melding
string<-"Heisann du"
a=5
for(i in 1:a){
  print(i)
}
bla
blla
```

A.1.2 C++

```
#include<stdio.h>
#include<iostream>
// A comment
int main(void)
{
  for(int i=0;i<5;i++){
     cout<<i;
  }
  printf("Hello World\n");
  return 0;
}
```

Bibliography

Aloui, C. and Mabrouk, S. (2010). Value-at-risk estimations of energy commodities via long-memory, asymmetry and fat-tailed garch models. *Energy Policy*.

Atchadé, Y., Fort, G., Moulines, E., and Priouret, P. (2011). Adaptive markove chain monte carlo: Theory and methods. In *Baysian Time Series Models*. Cambrudge Univeristy Press.

Box, G. E. P. and Muller, M. E. (1958). A note on the generation of random normal deviates. *The Annals of Mathematical Statistics*.

Christoffersen, P. F. (1998). Evaluating interval forecasts. In *International Economic Review*, volume 39, pages 841–862. Economics Department of the University of Pennsylvania.

Coles, S. (2001). *An Introduction to Statistical Modeling of Extreme Values*. Springer.

Eddelbuettel, D., Francois, R., Allaire, J., Ushey, K., Kou, Q., Bates, D., and Chambers, J. (2016). *Rcpp: Seamless R and C++ Integration*.

Fernandez, C. and Steel, M. F. T. (1998). On bayesian modeling of fat tails and skewness. *Journal of the American Statistical Assosiation*, 93(441).

Gamerman, D. and Lopes, H. F. (2006). *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*. Champman and Hall/CRC, 2nd edition.

Gelman, A., Roberts, G. O., and Gilks, W. R. (1996). Efficient metropolis jumping rules. In *Bayesian Statistics 5*.

Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Giot, P. and Laurent, S. (2003). Market risk in commodity markets: a var approach. In *Energy Economics*, volume 25, pages 435–457. Elsevier.

Givens, G. H. and Hoeting, J. A. (2013). *Computational Statistics*. John and Sons, 2nd edition.

Kupiec, P. H. (1995). Techniques for verifying the accuracy of risk measurement models. *Journal of Derivatives*.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*.

Müller, P. (1991). A generic approach to posterior integration and gibbs sampling. Technical report, Department of Statistics, Purdue University.

Næss, A. and Gaidai, O. (2009). Estimation of extreme value from sampled time series. In *Structural Safety*, volume 31, pages 325–334.

Næss, A., Gaidai, O., and Karpa, O. (2013). Estimation of extreme value by the average conditional exceedance rate method. *Journal of Probability and Statistics*.

Roberts, G. O. and Rosenthal, J. S. (2007). Coupling and ergodicity of adaptive markov chain monte carlo algorithms. *Journal of Applied Probability*.